Parallel Genetic Algorithm Implementation in Multidisciplinary Rotor Blade Design

Jongsoo Lee and Prabhat Hajela

Mechanical Engineering, Aeronautical Engineering and Mechanics Department

Rensselaer Polytechnic Institute

Troy, NY 12180

Abstract

The present paper describes an adaptation of genetic algorithms in the design of large-scale multidisciplinary optimization problems. A hingeless composite rotor blade is used as the test problem, where the formulation of the objective and constraint functions requires the consideration of disciplines of aerodynamics, performance, dynamics, and structures. A rational decomposition approach is proposed for partitioning the large-scale multidisciplinary design problem into smaller, more tractable subproblems. A design method based on a parallel implementation of genetic algorithms is shown to be an effective strategy, providing increased computational efficiency, and a natural approach to account for the coupling between temporarily decoupled subproblems. A central element of the proposed approach is the use of artificial neural networks for identifying a topology for problem decomposition and for generating global function approximations for use in optimization.

Introduction

An important area of research which has received considerable recent attention, is the development of optimization methodology applicable to large-scale multidisciplinary systems [1,2]. The initiative has been largely motivated by a recognition that the design and development of a complex engineering system can no longer be conducted by handling its different subsystems in isolation. Design synthesis in multidisciplinary systems is typically characterized by a large number of design variables and constraints; additionally, there are complex interactions between the participating subsystems which must be both identified and then suitably represented in the design process. In a number of problems of interest, the design space itself may be multimodal, thereby introducing the need for a global search strategy which offers an increased probability of locating the global optimum. The latter may contribute to demands on computational resource requirements, and the need exists for further development of function approximation methods that alleviate these requirements. The analysis and design of a helicopter rotor blade is an intrisically coupled multidisciplinary problem - strong interactions exist between disciplines of aerodynam-

---
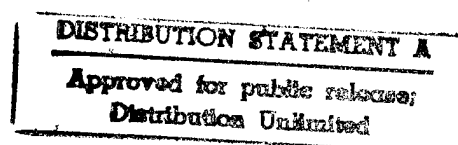
ics, acoustics, dynamics, and structures [3]. Design optimization with the proper consideration of couplings among disciplines can provide significant potential to improve rotor blade design performance, and formal optimization techniques have been explored extensively in rotorcraft design problems [4-8]. Recent research has focused on the application of advanced design techniques to tailor the rotor blade properties and geometry for multidisciplinary design requirements; the tailored blade response reduces the fixed system hub loads, results in enhanced aeroelastic stability, or improved aerodynamic performance measured in terms of power required in different flight regimes.

The design space for this rotor blade design problem is nonconvex, is characterized by a mix of continuous, discrete and integer design variables, and has an underlying analysis which is inherently nonlinear and computationally demanding. A number of previous studies have indicated that the use of traditional mathematical programming based optimization techniques may result in a suboptimal design. Furthermore, the ability to include discrete variables in these traditional methods requires the use of specialized techniques such as the branch-and-bound [9], which is only effective for moderate sized problems. Genetic algorithms (GA's) have received considerable recent attention in the structural optimization problems [10-12]. These methods belong to a category of stochastic search techniques which use random sampling to conduct a highly exploitative search. The distributed nature of this search makes it suitable for search in a nonconvex or disjoint design space [13]; furthermore, it is naturally amenable to handling discrete and integer variables in the search process. GA's have been applied to the design of rotorcraft blades for reduced vibration [14]. Specialized strategies have been proposed to extending the use of GA's [15] in large dimensionality problems.

Decomposition methods [16,17] have emerged as an alternative solution strategy to large-scale design problems. Here the optimal solution to the design problem is obtained as a number of coordinated solutions of smaller subproblems; solution coordination is necessary to account for any interactions among the decomposed subproblems. In this approach, the number of design variables in each subproblem can be kept small, and, furthermore, decompositions along the lines of disciplines may be possible in some situations. Given the nature of the design space for the rotor blade design problem, GA's must be considered as a solution strategy of choice, and their adaptation in the decomposition-based approach is the subject of the present paper. A preliminary study along such lines, with a small number of design variables and constraints, is reported in [18].

In using GA's in a decomposition-based design environment, each subproblem is assigned a subset of design variables and those constraints which are most critically affected by the variables. These smaller sized subproblems can be handled by the genetic algorithm without any specialized treatment, if the interactions between the temporarily decoupled systems are appropriately considered. The challenge in the approach resides in developing a rational procedure for determining the topology of decomposition, and in a procedure that naturally accounts for the interactions among the decomposed subproblems. The present paper outlines an approach whereby causality relations developed through the use of neural networks [19] are used to facilitate the task of problem decomposition. Once the subproblems are generated, GA based searches are conducted in parallel in each of the subproblems. The paper describes strategies by which changes in a subproblem are communicated to other subproblems through inter-population migration of designs.

It should be noted that such an approach allows for parallel processing of multiple populations, adding to the computational efficiency of the genetic search process. As noted earlier, the computational costs involved in computing vibratory loads and power components for the rotor blade problem are significant, as these computations entail a nonlinear and time-dependent analysis. A neural network based function approximation approach is used in the present work, where both the multilayer back-propagation network (BP network) [20] and a variant of the counter-propagation network (CP network) [21] are used for this purpose. The pattern completion capabilities of the CP network are shown to be particularly applicable in the decomposition-based design environment. Subsequent sections of the paper describe a typical multidisciplinary rotor blade design problem, encompassing various facets of problem complexity - coupling, presence of discrete/ integer variables, and high cost of analysis. The approach for parallel GA implementation in a decomposition-based design is discussed and applied to the rotor blade design problem. A comparison of optimization results obtained from the proposed approach, and an all-in-one (non-decomposition-based) approach is also presented.

Decomposition Methods in Multidisciplinary Design

Decomposition-based design methods have been proposed as a solution to large-scale coupled problem, wherein the original problem is decomposed into a set of smaller, more tractable subproblems. The ability to create smaller subproblems which represent the full complexity of the original problem may allow for parallel processing of solutions, and contribute to a better understanding of the problem domain. Traditional decomposition strategies fall into three principal categories - object decomposition, aspect decomposition, and sequential decomposition [22]. Object decomposition partitions a system into physical elements of the system (structural dynamic system can be represented by mass, damper, and spring.), aspect decomposition is based on the partition of a system mainly according to aspects of physics of the system (aircraft design may require aerodynamics, propulsion, dynamics/control, and structures disciplines.), and sequential decomposition is applied to problems dependent on the flow of design information. The structure of typical design problems after decomposition may either be hierarchical or non-hierarchical, depending upon the coupling between the decomposed subproblems. The coupling may be hierarchical, where it is possible to identify distinct tree-like patterns of interaction or, no obvious hierarchy may exist and there may be multiple one or two-way couplings among subsystems (Fig.1). Decomposition-based formal optimization methods have been applied to the design of hierarchical or non-hierarchical systems [23,24]; these studies have focused on methods to account for couplings between the subproblems.
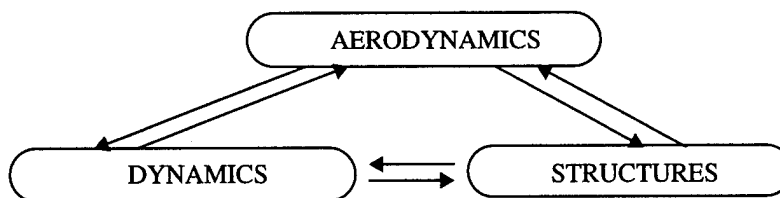

Fig.1 Nonhierarchical system interaction

In using decomposition principles in multidisciplinary design optimization, an issue that has to be considered at the very outset is the search for a rational approach to decompose the problem. The identification of optimal problem decomposition has been studied to improve computational efficiency and reliability in engineering design problems [25,26]. The implementation of an optimal decomposition results in weakly connected structures, and provides a more tractable design environment. A sequential decomposition of the design process has been proposed in concurrent engineering applications [27], where the branch-and-bound algorithm was used to identify overlapping design variables. Hypergraphs have been used to represent design optimization models, and the optimal decomposition of design problem has been formulated as a hypergraph partitioning problem based on a graph theory [28].
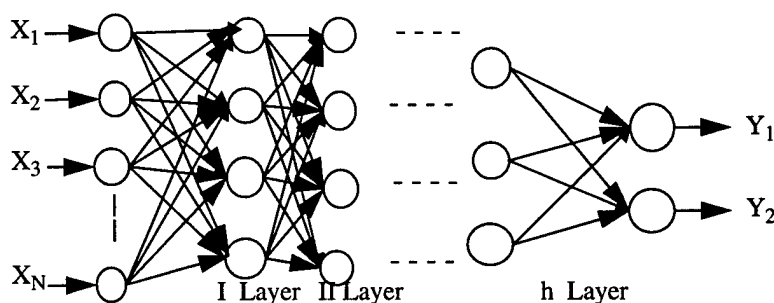

Fig.2 A BP network with h-hidden layers

Topology of Problem Decomposition

Previous research [19] has established that neural networks can be used to identify dependencies among design variables and design objectives, and to provide a guideline for problem decomposition. A BP network (see Fig. 2) can be trained using a set of input-output training patterns distributed uniformly over a problem domain. Network training involves the selection of the interconnection weights between the neurons, so that for each of the input patterns, the error between the network predicted output and the actual known output is minimized. Detailed information on the BP network, and on the mechanics of training are available in [29]. Once trained, the interconnection weights can be analyzed to determine the importance of any input component on an output quantity of interest. Such weight analysis can be represented by two ways - causal relations between input and output quantities in an absolute sense that reflects only the magnitude of the weights, or in a normalized sense that additionally preserves the signs of the weights. Both approaches result in the formation of a transition matrix [T], the components T_{ij} of which reflect the importance of the i-th input quantity on the j-th output component. In the first approach (ABS), the dominance is only indicated in a qualitative sense; the second approach (ALT), used in the present work is described in greater detail in [30]. In this approach, the matrix product of the interconnection weight matrices is performed as indicated in Eq. (1), and the elements of the transition matrix normalized as shown in Eq. (2).

$$[T] = \prod_{k=1}^{N-1} W^k \tag{1}$$

$$\overline{T}_{ij} = \frac{T_{ij}}{max_i \left| T_{ij} \right|} \tag{2}$$

In the above, W^k is the k-th weight matrix, the coefficients W_{ij}^{kl} of which represents the interconnection weight between the i-th neuron of the k-th layer and the j-th neuron of the l-th layer; N denotes the total number of layers of neurons in the network architecture. This normalized matrix $\overline{T}_{ij}$ incorporates the effect of the sign of interconnection weights in the analysis. Once this quantitative causal relationship is established, the problem can be optimally partitioned as described in the following section.

Optimal Partitioning for Problem Decomposition

The neural network based weight analysis provides valuable information about the extent of couplings in the multidisciplinary system. In order to implement a decomposition-based design strategy, the system must be partitioned into an appropriate number of subproblems depending on available computing machines or parallel processors. Optimal partitioning schemes for system decomposition have been widely used in design and manufacturing applications for process and scheduling [27]. A reasonable and logical approach for partitioning is one where balanced subsets of design variables would be assigned to different subproblems, and where each subproblem would be responsible for meeting the system level design objectives and for satisfying constraints most critically affected by the design variables of that subproblem. This approach is implemented in the context of the neural network based transition matrix as follows.

For a transition matrix [T] obtained from a trained BP network, partition this matrix into K (where $2 \leq K \leq NCON$, and NCON is the total number of constraints in the design problem) different groups denoted as G_k; each group contains design variables x_i which have the strongest influence on constraints belonging to the group G_k. To formalize the partitioning procedure, define a grouping identification matrix with elements V_{ij} as follows.

$$V_{ij} = 1 \quad if \quad x_i \in G_k$$
$$V_{ij} = 0 \quad if \quad x_i \notin G_k \tag{3}$$

In the above, the subscript 'j' refers to the j-th constraint. To obtain an optimal partitioning, a performance index PI is determined as follows;

$$PI = \frac{1}{\alpha} \sum_{i=1}^{NDV} \sum_{j=1}^{NCON} \frac{A_{ij}}{z_i} \tag{4}$$

where,

$$A_{ij} = V_{ij} \left| T_{ij} \right| \tag{5}$$

$$\alpha = \sum_{i=1}^{NDV} \sum_{j=1}^{NCON} \left| T_{ij} \right| \tag{6}$$

and, NDV is the number of design variables. Note that the elements A_{ij} in Eq. (4) are obtained as a scalar product in Eq. (5), and have a value of either zero or the absolute value of the coefficient T_{ij} of the transition matrix; z_i is the number of non-zero values in the i-th row of the matrix [A]. The mathematical statement of optimal problem partitioning can be formulated as follows.

Minimize

$$f = \frac{1}{PI} + \frac{1}{2} \sum_{i=1}^{K} \sum_{j=1}^{K} (1 - \delta_{ij}) |Nx_i - Nx_j| \tag{7}$$

subject to

$$1 \leq N(g_j)_{G_k} \leq N(g_j)^U \tag{8}$$

where, δ_{ij} is a Kronecker delta, Nx_k is the number of design variables assigned to group G_k, and $N(G_j)_{G_k}$ is the number of constraints in group G_k. The objective function has two components - the first term leads to a maximization of the performance index P I, while the second term ensures a minimal difference between the number of design variables in each group. The constraint in Eq. (8) is necessary to limit the number of design constraints allocated to a specific group. An upper bound on the number of constraints denoted as $N(g_j)^U$ was used in this problem. In this optimization problem, the design variables are the allocation of elements of the grouping identification matrix V_{ij}. This is an integer programming problem which is conveniently solved using the GA approach.

Genetic Algorithms in Multidisciplinary Design Optimization

The GA approach is based on representing possible solutions to a given problem by a population of bit strings of finite lengths, and to use transformations analogous to the biological reproduction and evolution to improve and vary the coded solutions. A commonly used approach is to represent each design variable value by a fixed length of binary string. The binary string representations for each variable can be placed head-to-tail to create a chromosome-like representation of the design. Several such chromosomal strings are defined to constitute a population of designs, which includes a mix of feasible and infeasible designs. This population of designs is then subjected to three basic genetic transformations referred to as reproduction, crossover, and mutation [31]. These transformations are applied selectively to those members of the population which are deemed more fit than others; in an unconstrained function maximization problem, the objective function could be used as a fitness function. This bias in the selection process allows the more fit designs to contribute their characteristics to subsequent generations, and eliminates the less fit designs from the population. In large-scale rotor blade design problem, the string representations of designs can get quite long, and the number of design alternatives in the search space which must be examined also increases dramatically. To circumvent the need for increased computational resources when using GA's in large scale problems, a decomposition-based approach was implemented; details of this implementation, including the approach to account for subproblem interactions, is described in the following section.

Coordination Strategies in Parallel GA Implementation

Consider the design problem to be formulated in terms of a design variable vector X. Also, let the design constraints $g_j(X)$ belong to the global constraint set G. The vector X and constraint set G are said to define a system level problem. Assume further that the best topology for decomposing the problem domain was established through an optimal partitioning scheme described in the previous section, and that three subproblems A, B, and C were established. Note that the number of partitioned groups was chosen as K=3 in this study. The design variables and constraints for each of these subproblems are denoted by X_A, X_B, X_C, and g_A, g_B, and g_C, respectively. The objective function $F(X)$ for each of the subproblems is the same, and is the system level objective function.

$$Min \quad or \quad Max \quad F(X)$$

$$subject \ to \ G \equiv \ \{g_j(X), \ j \ = \ 1...NCON\} \ \leq 0 \tag{9}$$

After the optimal problem partitioninig, the design optimization problem is represented by the following three subproblems in Eq. (10).

$$Min \quad or \quad Max \quad F(X_A), \quad subject \ to \ g_A(X_A) \ \leq 0, \quad X_B, \ X_C \ = \ const$$

$$Min \quad or \quad Max \quad F(X_B), \quad subject \ to \ g_B(X_B) \ \leq 0, \quad X_A, \ X_C \ = \ const \tag{10}$$

$$Min \quad or \quad Max \quad F(X_C), \quad subject \ to \ g_C(X_C) \ \leq 0, \quad X_A, \ X_B \ = \ const$$
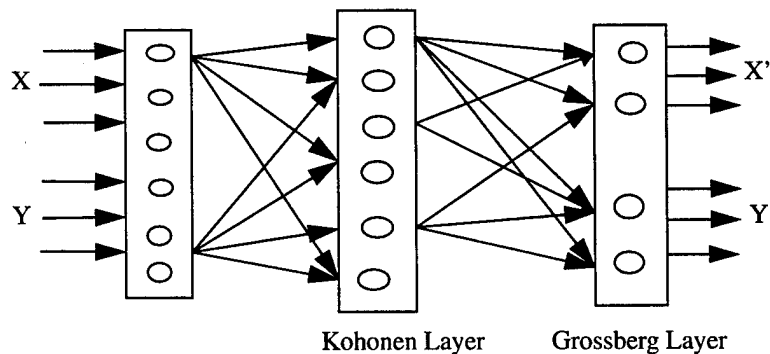

Fig.3 Schematic representation of a CP network

The GA strategy can be implemented for each of the subproblems; shorter string lengths, and hence smaller population sizes are required in each subproblem. The genetic evolution process can be carried out in parallel. The principal difficulty in this approach is that the constraint sets identified for a particular subproblem, are not completely independent of the design variables that may have been assigned to another subproblem. Such coupling must be accommodated in the parallel optimization scheme, and was faciliated through the use of a CP network based approximation. The architecture of this network (shown in Fig. 3) consists of two primary

layers; the first layer referred to as the Kohonen layer simply classifies a given input vector as belonging to a particular category, while the second layer (the Grossberg layer) develops an average output for each category of inputs. In a modification of this basic architecture [21], an input vector was first classified as belonging to more than one category, albeit to different degrees, and a nonlinear weighted average of outputs corresponding to dfifferent categories developed as the network output. An important property of this network is a pattern completion capability - if an incomplete input pattern is presented to the network, the network estimates the most likely make-up of the missing components.

In the present work, the GA based optimizer in each subproblem was linked to a trained CP network. The inputs to the CP network in each subproblem were the design variables for that subproblem, and approximations of the best combinations of variables in other subproblems. In the preliminary testing of the CP network [32], it was shown that the quality of function approximations was significantly enhanced if approximations of variables belonging to other subproblems were provided as opposed to the case where these components were altogether eliminated from the input vector. In the present implementation, the best designs from each subproblem were migrated to all other subproblems after a prescribed number of cycles of GA based search. A schematic of this set-up is shown in Fig. 4. A stepwise description of the numerical process is as follows.

a. Develop a trained CP network to map the relation; $X \rightarrow \{G,F\}$ where, F is the system level objective function.

b. Develop a trained BP network to map the relation; $X \rightarrow \{G\}$.
    Analyze the weights of this trained network to establish a topology for decomposition. Note that this step can be replaced by a heuristically determined decomposition approach.

c. For each subset of design variables in a subproblem, initialize a starting population of designs. Denote design variables of other subproblems as problem parameters for the sub-problem under consideration.

d. Evolve each subproblem in parallel for a fixed number of generations. Function analyses in each subproblem is obtained by presenting subproblem design variables and problem parameters to trained CP network of step (a).

e. Conduct inter-population migration of problem parameters. Two forms of this updating were implemented:
    S1: For each subproblem, use as problem parameters the current best design variable values of other subproblems.
    S2: For each subproblem, evaluate all possible combinations of problem parameters. These would include current problem parameters, and those available as the new best designs in other subproblems. Select a combination so that the current objective function either improves or, at worst, stays the same.

f. Repeat from step (d) until no further improvement in the objective function value is obtained or the allowed number of function evaluations have been performed.
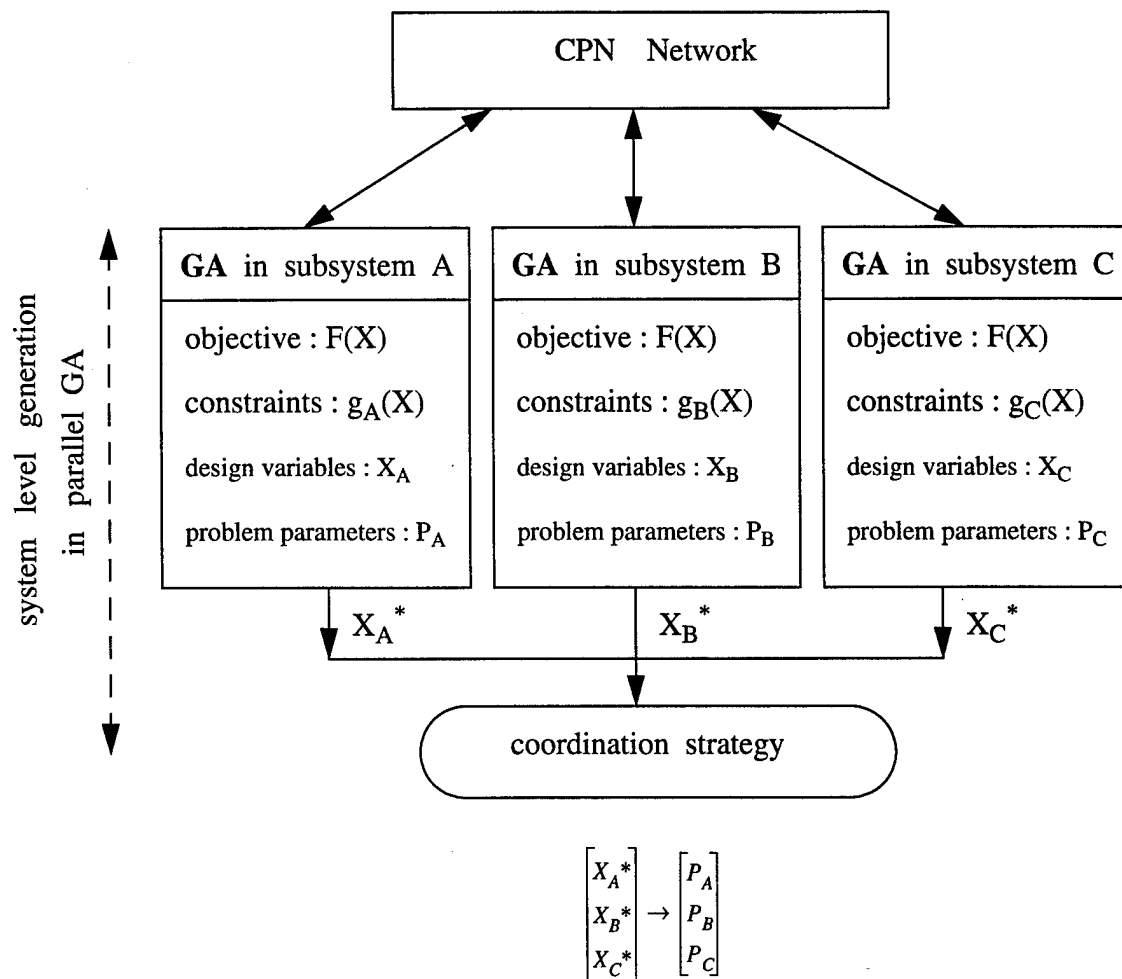
Fig.4   Parallel design process in GA based decomposition approach

Multidisciplinary Design of a Helicopter Rotor Blade

The use of composites in rotor blades has opened up new possibilities for enhanced structural, aerodynamic, and dynamic performance. These materials allow for the fabrication of non-rectangular blades with variations in twist distribution and airfoil sections along the blade span, thereby contributing to increased flexibility in aerodynamic design. Satisfactory aerodynamic design requires that the required horsepower for all flight conditions not exceed the available horsepower, that the rotor disk must retain lift performance to avoid blade stall, and that the vehicle remain in trim. Important factors in structural design include material strength considerations for both static and dynamic load conditions. A combination of flapwise, inplane, torsion, and centrifugal forces typically comprise the static loading. Another important consideration that encompasses both structural and aerodynamic design, is the autorotation capability. The autorotation requirement pertains to maintaining the mass moment of inertia of

the rotor in the rotational plane at an acceptable level. This is a function of the vehicle gross weight, rotor aerodynamic performance, and the rotor system mass moment of inertia. Finally, dynamic design considerations of the rotor blade pertain to the vibratory response of the blade under the applied loads; this design limits the dynamic excitation of the fuselage by reducing the forces and moments at the blade root.

A finite element in time and space formulation was used to model the dynamics of the blade [33]. This formulation is based on a multibody representation of flexible structures undergoing large displacements and finite rotations, and requires that the equations of motion be explicitly integrated in time. An unsteady aerodynamic model was used to obtain the induced flow and to calculate the aerodynamic forces and moments in hover and forward flight.
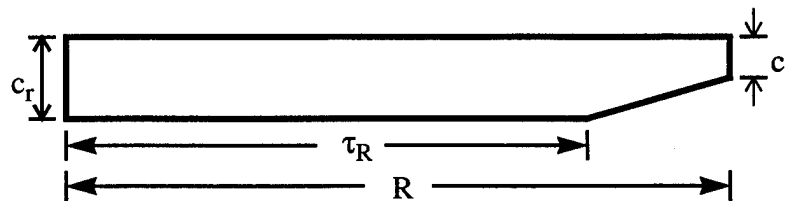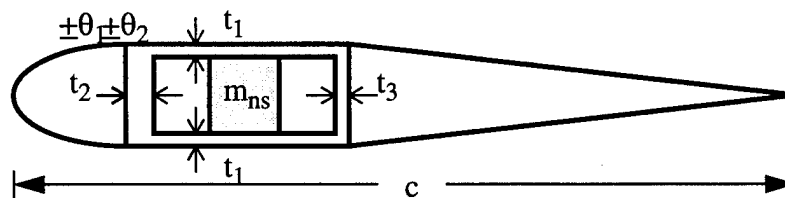
Fig.5a Rotor blade planform geometry

Fig.5b Cross section of blade airfoil

Design Model

The objective of the design problem is to design the blade geometry and internal structure to minimize a weighted sum of the rotor fixed system hub shear force and bending moments for a hingeless rotor blade in forward flight; aerodynamic, performance and structural design requirements are considered as constraints, and dynamic requirements constitute a multicriterion objective function. The premise behind the approach is that a minimization of the hub loads and moments translates into lower vibrations which are transmitted to the fuselage structure. The design variables used in the multidisciplinary rotor blade design problem are shown in Figs. 5a-5b. The blade is divided into 10 segments along the spanwise direction. As shown in Fig. 5b, each segment is defined by three cross-sectional dimensions of the thin-walled composite box-beam. In addition, there are 5 nonstructural tuning masses along the span, and 2 ply orientations θ_1 and θ_2. Finally, the geometry of the blade was defined by a blade twist distribution parameter, a chord ratio, and a spanwise position of blade taper inception. The rotor angular velocity was also considered as a design variable resulting in a total of 42 design variables. The lower and upper bounds on the

design variables are shown in Table 1. Design constraints in the problem include power required in hover and forward flight, and denoted as HP_h and HP_f, respectively, the figure-of-merit, η, which reflects the power performance ratio in hover out-of-ground effect, autorotational index, AI, lift performance, C_T/σ, blade weight, W_b, local buckling stresses in the structural box sections, σ_{buck}, and a composites failure measure, $\overline{R}$. This resulted in a total of 20 design constraints.

The analysis model to compute the blade response as a function of the design variables is computationally intensive, and in order to be used in conjunction with a GA based search strategy, requires an approximate analysis capability. As described in a previous section, the CP network provides a global approximation strategy wherein a trained network provides an approximation of the output for a design alternative considered by the GA. It should be noted that given the distributed nature of GA based search, traditional Taylor series approximations have little practical value.

Results and Discussion

A number of numerical experiments were conducted to determine the validity of the proposed approach. At the very outset, BP and the CP networks were trained to develop causal relations and to generate function approximations. With a total of 1,450 training patterns, the BP network yielded maximum errors of 4% when presented with 100 input patterns which were not part of the training process. Average errors in this case were only about 3%. The CP network required a larger number of training patterns - a total of 12,800 patterns were actually used to get maximum errors of 7% in the predictive capability. Again, the average errors here were only of the order of 5%. More detailed discussions on the use of neural networks to model rotor blade response are available in [32].

An analysis of the interconnection weights of the trained BP network was performed to determine the topology for problem decomposition. Problem decomposition by constraints requires a transition matrix generated from the weights of a neural network, trained to develop the mapping between design variables as inputs and the corresponding constraint values as output. The ALT approach was selected for this purpose as previous studies [18,32] indicated that it was more effective in large-scale design problems. A total of only seven constraints was considered in the optimal problem partitioning, as constraints for local buckling were arbitrarily assigned to the same set of design variables which were driving the structural weight constraints. On the basis of this transition matrix, the problem decomposition was obtained using the optimal partitioning scheme described in Eqs. (7)-(8). For the chosen value of K=3, the GA based optimal partitioning result is shown in Table 2. Each subproblem consists of design variables and constraints which have strong influences each other. Note that in any row of this matrix, the magnitude of coefficients inside each solid box defining a subproblem are, in general, higher than those outside of this box. The second term in Eq. (7) also has a strong influence on the partitioning, resulting in each group being assigned as the same number of design variables. Interestingly, the optimization results in a grouping of constraints which have similar properties; HP_h, η, and AI are properties in hover condition, and HP_f and C_T/σ are calculated from forward flight condition. Table 3 shows the suggested decomposition topology, where the system level design variable vector X and con-

straint set G are broken down into subsets with respect to design constraints. In Table 3, m^i is the tuning mass, t_j^i is the j-th flange of the thin-walled beam in the i-th segment, θ_t and δ are the blade twist angle at the tip and the shape parameter defining the nonlinear distribution of blade twist, respectively, λ_c is a chord ratio, τ_R is a nondimensionalized value of taper inception point, Ω is a rotational speed, and $\pm\theta_i$ is a composite ply angle in the box-beam.

Table 1 Range of Design Variables Used in Neural Network Training

design variable	symbol	X_i	min	max	precision
tuning mass	m^i	X_1 - X_5	0.0	4.0	0.1
horizontal flange thickness ratio	t_1^i	X_6 - X_{15}	0.25	0.35	0.01
left vetical flange thickness ratio	t_2^i	X_{16} - X_{25}	0.20	0.30	0.01
right vertical flange thickness ratio	t_3^i	X_{26} - X_{35}	0.20	0.30	0.01
blade twist [deg]	θ_t	X_{36}	4.0	8.0	0.1
twist shape parameter	δ	X_{37}	0.3	1.0	0.1
taper inception point	τ_R	X_{38}	0.50	0.80	0.01
chord ratio	λ_c	X_{39}	1.0	2.0	0.1
rotational speed [rad/sec]	Ω	X_{40}	24.0	30.0	1.0
ply angle of inner vertical flange [deg]	θ_1	X_{41}	30	90	5
ply angle of outer vertical flange [deg]	θ_2	X_{42}	90	90	5

The decomposition-based design was implemented for a number of test cases, using the parallel GA approach described in previous sections. Recognizing the random nature of the GA search, the search process was repeated for a number of different settings for the pseudo-random number generator. GA parameters such as probabilities of crossover and mutation, population size, and string lengths for the design, are summarized in Table 3.

The convergence histories of the system level objective function for two different strategies of the coordination are shown in Fig. 6. In strategy S1, the solution exhibits significant oscillations due to introduction of infeasibilities with each update of problem parameters. However, the degree of oscillation tends to decrease as the solution converges. Numerical experiments with different random numbers demonstrate that this approach yields similar best objective function values after a few executions of the search process. The results for strategy S2 show a monotonically decreasing value of the system level objective function (this was a requirement of the updating scheme); however, the best objective function value appeared to be heavily dependent upon the initialization of the radom number generator. Strategy S1 consistently resulted in better values of the best objective function value than strategy S2. However, strategy S2 does guarantee that once a feasible design is identified, an abrupt termination of the search process will at least produce a feasible design. The results of the decomposition based strategy were compared against those obtained from treating all design variables and constraints in a single group by an all-in-one approach. Figs. 7 (a)-(b) show this comparison in terms of the best objective function value and the required CPU time. Here, six different sets of results are presented, including four different implementations of the all-in-one approach and two sets of results corresponding to decomposition-based strategies S1 and S2. The all-in-one strategies include a straightforward GA implementation, or

the plain GA, GA with directed crossover, GA with multistage strategies, and a combination of the directed crossover and multistage search [15].

Table 2 Optimal partitioning of system decomposition

	HP_h	F M	A I	HP_f	C_T/σ	W_b	$\overline{R}$
x 1	-0.31122	-0.20702	-0.10739	0.01432	-0.02469	-0.31651	-0.04328
x 2	-0.31767	-0.22166	-0.09736	0.02230	0.00049	-0.32416	0.15529
x 3	-0.33205	-0.27008	-0.05503	0.03852	0.06996	-0.34035	0.24476
x 9	-0.06477	-0.02082	-0.03332	-0.02082	-0.01283	-0.06346	0.02599
x10	-0.08221	-0.04840	-0.02593	-0.00124	0.05655	-0.08428	0.02478
x11	-0.10063	-0.06934	-0.01222	0.01204	-0.00474	-0.10248	0.11756
x18	0.01140	0.01592	-0.00378	-0.02108	-0.06511	0.01001	-0.03155
x21	-0.01210	-0.03661	-0.00413	-0.00026	0.02378	-0.01415	0.03511
x24	-0.00805	-0.00719	0.00293	0.01389	-0.06470	-0.00944	-0.08279
x31	0.02001	0.04601	-0.00441	-0.00455	0.00017	0.01652	0.03500
x33	-0.02104	-0.01701	0.00217	0.01078	0.01453	-0.02130	0.04315
x34	-0.01504	-0.03227	-0.00062	-0.01021	0.01551	-0.01440	-0.01239
x38	-0.20777	-0.17080	1.00000	0.15572	0.10275	-0.22482	0.37426
x41	0.03738	0.03629	-0.00408	-0.02273	-0.03160	0.00321	0.06405
x 8	-0.07918	-0.00643	-0.03324	-0.02226	-0.03423	-0.07776	0.01929
x15	-0.05017	-0.04212	0.01513	-0.01349	-0.09035	-0.04858	0.03682
x19	0.00161	-0.00197	-0.00065	0.00359	-0.02500	-0.00144	-0.06336
x20	-0.01344	-0.01270	-0.00095	-0.01467	-0.04590	-0.01014	0.00908
x22	0.01889	0.03187	0.00276	-0.01515	-0.04468	0.01751	0.01327
x23	0.00462	0.02569	-0.00061	-0.02368	-0.10400	0.00149	-0.01317
x27	0.00739	0.02657	-0.00967	-0.00698	-0.09053	0.00595	-0.13789
x28	0.02278	0.02255	0.00411	0.03776	-0.08852	0.02164	-0.04513
x30	-0.02402	-0.05278	-0.00852	0.02067	-0.04556	-0.02560	0.08914
x32	0.00829	0.01154	0.00103	-0.00334	-0.06217	0.00558	-0.05512
x35	-0.02667	-0.05944	-0.00075	0.00296	0.00125	-0.03032	-0.03318
x36	0.03934	0.03059	-0.00346	1.00000	1.00000	0.00344	0.14198
x37	0.01964	0.01195	-0.00840	-0.44560	-0.45198	0.00162	-0.02298
x42	0.01018	-0.00509	0.00175	0.04040	-0.07911	0.00112	-0.06664
x 4	-0.29512	-0.17174	0.00144	0.03140	0.07893	-0.30238	0.22224
x 5	-0.30075	-0.17085	0.12103	-0.01062	-0.04009	-0.30571	0.40037
x 6	-0.05802	-0.02470	-0.02052	0.01774	-0.01167	-0.05586	-0.30023
x 7	-0.13262	-0.08277	-0.03034	0.05067	0.04350	-0.12982	-0.37191
x12	-0.13337	-0.13914	-0.00130	-0.00175	0.02686	-0.13284	0.13375
x13	-0.09454	-0.04700	0.01197	0.00952	0.13411	-0.09813	0.22234
x14	-0.10299	-0.05171	0.02424	-0.01474	-0.01785	-0.10358	0.14926
x16	-0.02061	-0.02528	-0.00536	0.00044	0.07328	-0.01802	0.00577
x17	-0.05248	-0.06647	-0.00443	0.01262	-0.06872	-0.00707	-0.09438
x25	-0.00733	-0.02811	0.00606	0.01262	-0.06872	-0.00707	-0.09438
x26	-0.05582	-0.04970	-0.00349	0.01341	0.10149	-0.05465	0.02636
x29	-0.00498	-0.02420	0.00076	0.00700	-0.02614	-0.00261	-0.05519
x39	1.00000	0.21680	-0.14504	-0.33351	-0.21798	1.00000	-0.82832
x40	0.01542	1.00000	0.40144	0.77140	-0.07304	0.00535	1.00000

Fig. 7(a) shows the results for three different initializations of the random number generator. The best objective function values show the expected improvement in performance as advanced GA strategies are used. The decomposition-based design strategy S1 gives the best overall result. Strategy S2 yields results that were similar to the plain GA implementation. Clearly, this updating scheme severely limits the exploration of the design space by requiring that problem parameter updates must maintain a monotonic convergence trend in the objective function value. In Fig.7(b), the relative CPU time required to reach the best objective function value obtained from the plain GA is presented for the different design strategies. This figure clearly demonstrates that the decomposition approach is much more efficient from a computational resource standpoint due to the parallel implementation in optimization. This figure also reinforces the fact that returns from a decomposition-based approach are much more significant as the size of the problem increases; the results for the 42 design variable problem are compared against similar results for a 14 design variable problem in [18]. For the best design obtained in the decomposition based approach, the optimal stiffness distribution of rotor blade box-beam is shown in Fig. 8.

Table 3   Topology of system decomposition

	subsystem [A]	subsystem [B]	subsystem [C]
objective	$F(X) = c_1 F_z + c_2 M_y + c_3 M_z$		
constraints	$HP_h \leq HP_a$ $\eta^L \leq \eta \leq \eta^U$ $AI \geq AI^L$	$HP_f \leq HP_a$ $C_T/\sigma^L \leq C_T/\sigma \leq C_T/\sigma^U$	$W_b \leq W_b^U$ $\overline{R} \leq 1$ $\sigma_{buck} \leq \sigma_{all}$
design variables	$m_1, m_2, m_3, t_1^4, t_1^5,$ $t_1^6, t_2^3, t_2^6, t_2^9,$ $t_3^6, t_3^8, t_3^9, \tau, \pm\theta_1$	$t_1^3, t_1^{10}, t_2^4, t_2^5, t_2^7,$ $t_2^8, t_3^2, t_3^3, t_3^5,$ $t_3^7, t_3^{10}, \theta_t, \delta, \pm\theta_2$	$m_4, m_5, t_1^1, t_1^2,$ $t_1^7, t_1^8, t_1^9, t_2^1, t_2^2,$ $t_2^{10}, t_3^1, t_3^4, \lambda_c, \Omega$

Table 4   GA control parameters used in decomposition approach

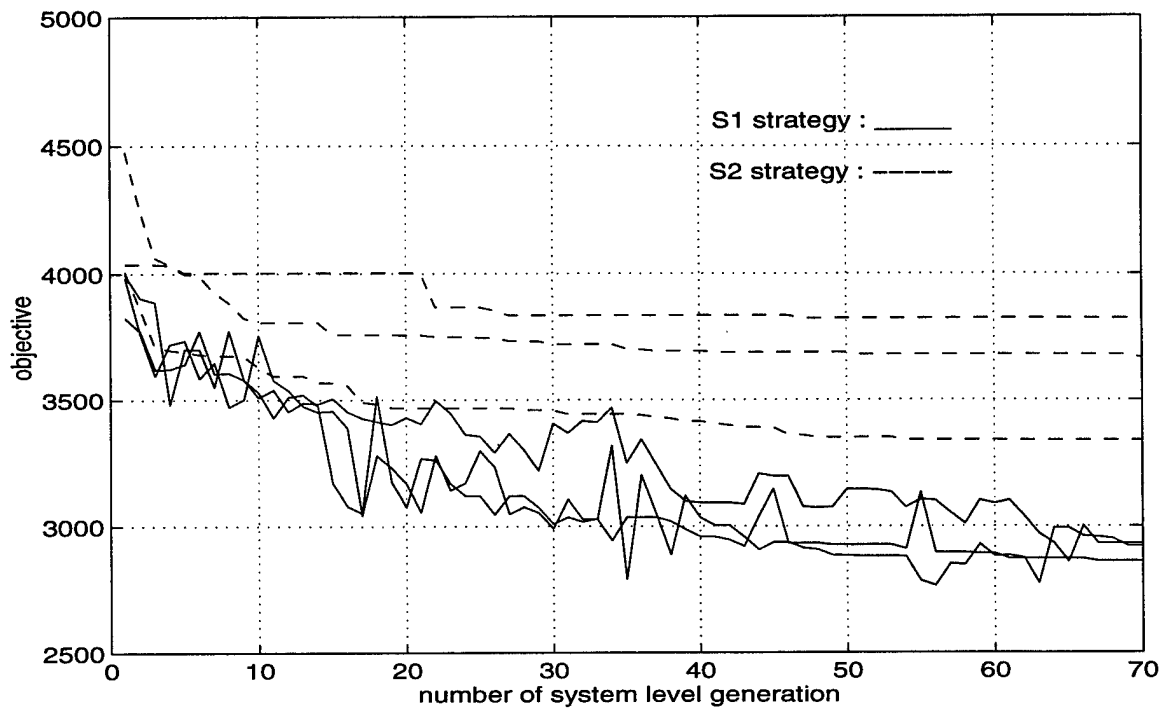|	subsystem [A]	subsystem [B]	subsystem [C]	all-in-one approach
$P_{crossover}$	0.80	0.80	0.80	0.80
$P_{mutation}$	0.02	0.02	0.02	0.01
NPOP	100	100	100	300
NSTRING	60	57	57	174

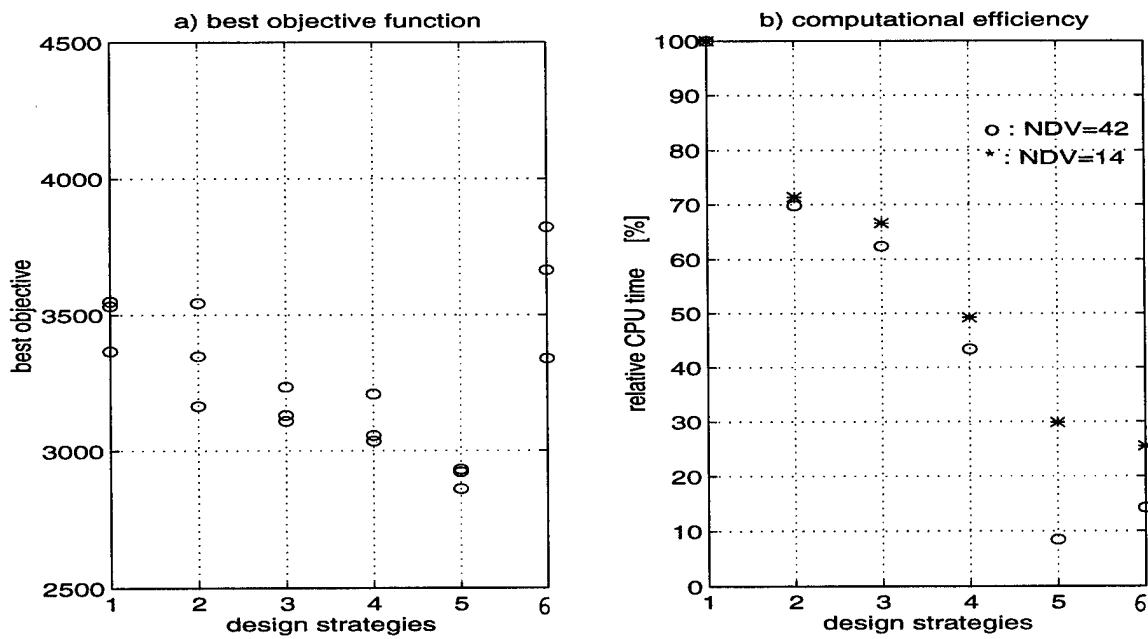Fig.6 Convergence history of design strategies in decomposition approach

Fig.7 Performance of large-scale design process; (1) plain GA, (2) directed crossover, (3) multistage search, (4) combined search, (5) strategy S1, (6) strategy S2
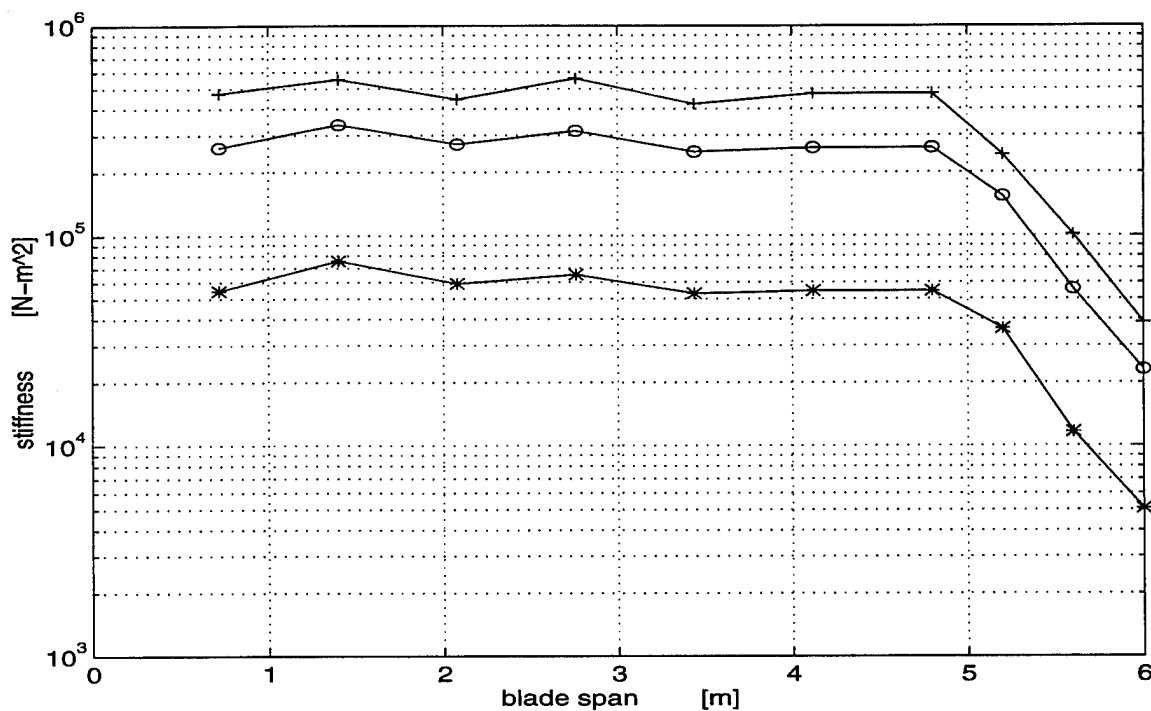
Fig.8   Optimal stiffness distribution along blade span; (+) lead-lag, (o) torsion, (*) flap

Concluding Remarks

The paper describes an approach for adapting genetic algorithms in the decomposition-based design of large-scale multidisciplinary systems. A primary focus of the research was in the development and implementation of a rational approach by which the multidisciplinary design problem could be partitioned into a number of balanced subproblems. This task of partitioning was formulated and solved as an optimization problems. This method used quantitative information about the dependence of system response to design variables extracted from trained neural networks in carrying out the optimal problem partitioning. Once the problem was decomposed, the GA based search was implemented in parallel in each of the subproblems. Strategies to account for the interactions among these decomposed subproblems was the other focus of the present study. The proposed methods were implemented in a multidisciplinary test problem - design of a helicopter rotor blade to minimize vibrations at the rotor hub. Numerical results show the effectiveness of the decomposition-based approach over traditional all-in-one strategies, as indicated by both better performance and lower computational resource requirements.

Acknowledgements

References

[1] Sobieszczanski-Sobieski, J., "Multidisciplinary Design Optimization: An Emerging New Engineering Discipline," World Congress on Optimal Design of Structural Systems, Rio de Janeiro, Brazil, August 2-6, 1993.

[2] Tolson, R. H., and Sobieszczanski-Sobieski, J., "Multidisciplinary Analysis and Synthesis: Needs and Opportunities," AIAA Paper No. 85-0584, 1985.

[3] Adelman, H., and Mantay, W. A. (eds), Integrated Multidisciplinary Optimization of Rotorcraft: A Plan for Development, NASA TM 101617, May 1989.

[4] Young, D. K., and Tarzanin Jr., F. J., "Structural Optimization and Mach Scale Test Validation of A Low Vibration Rotor," proceedings of the 47th Annual Forum of the American Helicopter Society, Phoenix, AZ, May 1991.

[5] Walsh, J. L., Bingham, G. L., and Riley, M. F., "Optimization Methods Applied to the Aerodynamic Design of Helicopter Rotor Blades," *Journal of American Helicopter Society*, Vol. 32, No. 4, pp. 39-44, October 1987.

[6] Chattopadhyay, A., Walsh, J. L., and Riley, J. L., "Integrated Aerodynamic/Dynanmic Opti mization of Helicopter Blades," NASA TM-101553, 1989.

[7] Friedmann, P. P., and Celi, R., "Structural Optimization with Aeroelastic Constraints of Rotor Blades with Straight and Swept Tips," *AIAA Journal*, Vol. 28, No. 5, pp. 928-936, 1990.

[8] Friedmann, P. P., "Helicopter Vibration Optimization Using Structural Optimization with Aeroelastic / Multidisciplinary Constraints," *Journal of Aircraft*, Vol. 28, No. 1, pp. 8-21, 1991.

[9] Land, A. H., and Doig, A. G., "An Automatic Method for Solving Discrete Programming Problems," Econometrica, 28, pp. 497-520, 1960.

[10] Hajela, P., and Lee, E., "Genetic Algorithms in Topological Design of Grillage Structures," Discrete Structural Optimization, pp. 30-39, Gutkowski, W., and Bauer, J. (eds.), Springer Verlag, Berlin, 1994.

[11] Le Riche, R., and Haftka, R. T., "Optimization of Laminate Stacking Sequence for Buckling Load Maximization by Genetic Algorithms," *AIAA Journal*, Vol. 31, No. 5, pp. 951-956, 1993.

[12] Rao, S. S., Pan, T.-S., and Venkayya, V. B., "Optimal Placement of Actuators in Actively Controlled Structures Using Genetic Algorithms," *AIAA Journal*, Vol. 29, No. 6, pp. 942-943, 1991.

[13] Hajela, P., "Genetic Search - An Approach to the Nonconvex Optimization Problem," *AIAA Journal*, Vol. 26, No. 7, pp1205-1210, July 1990.

[14] Hajela, P., and Lee, J., "Genetic Algorithm Based Sizing of Low Vibration Rotors," proceedings of the 5th ARO International Workshop on Dynamics and Aeroelastic Stability Modeling of Rotorcraft Systems, Troy, New York, October 18-20, 1993.

[15] Lin, C.-Y., and Hajela, P., "Genetic Search Strategies in Large Scale Optimization," proceedings of the 34th SDM Conference, La Jolla, California, 1993.

[16] Sobieszczanski-Sobieski, J., "Sensitivity of Complex, Internally Coupled Systems," *AIAA Journal*, Vol. 28, pp. 153-160, 1990.

[17] Bloebaum, C. L., and Hajela, P., "Heuristic Decomposition for Non-Hierarchic Systems," proceedings of the 32nd SDM Conference, Baltimore, MD, April, 1991.

[18] Hajela, P., and Lee, J., "Genetic Algorithms in Multidisciplinary Rotor Blade Design," proceedings of the 36th SDM Conference, New Orleans, LA, April 1995.

[19] Hajela, P., and Szewczyk, Z., "On the Use of Neural Network Interconnection Weights in Multidisciplinary Design," AIAA Paper No. 92-4808, proceedings of the 4th AIAA/NASA/ Air Force Symposium on Multidisciplinary Analysis and Optimization, Cleveland, Ohio, September 1992.

[20] Hajela, P., and Berke, L., "Neurobiological Computational Models in Structural Analysis and Design," *Computers and Structures*, Vol. 41, No. 4, pp. 657-667, 1991.

[21] Szewczyk, Z., and Hajela, P., "Neurocomputing Strategies in Decomposition Based Structural Design," proceedings of the 34th SDM Conference, La Jolla, CA, 1993.

[22] Wagner, T. C., and Papalambros, P. Y., "A General Framework for Decomposition Analysis in Optimal Design," Advances in Design Automation-1993, Gilmore, B. J. (ed.), Vol. 2, ASME, pp. 315-325, New York, 1993.

[23] Sobieszczanski-Sobieski, J., "A Linear Decomposition Method for Large Optimization Problems - Blueprint for Development," NASA TM 83248, 1982.

[24] Sobieszczanski-Sobieski, J., "Optimization by Decomposition: A Step from Hierarchic to Non-Hierarchic Systems," 2nd NASA/Air Force Symposium on Recent Advances in Multidisciplinary Analysis and Optimization, NASA CP-3031, Sep. 1988.

[25] Kusiak, A., and Chow, W. S., "Efficient Solving of the Group Technology Problem," *Journal of Manufacturing Systems*, Vol. 6, No. 2, pp. 117-124, 1987.

[26] Michelena, N. F., and Papalambros, P. Y., "A Network Reliability Approach to Optimal Decomposition of Design Problems," Advances in Design Automation-1994, Gilmore, B. J. (ed), Vol. 2, ASME, pp. 195-204, New York, 1994.

[27] Kusiak, A., and Wang, J., "Decomposition of the Design Process," *Journal of Mechanical Design*, Vol. 115, pp. 687-695, December, 1993.

[28] Michelena, N. F., and Papalambros, P. Y., "A Hypergraph Framework for Optimal Model-Based Decomposition of Design Problems," Technical Report UM-MEAM 95-02, Department of Mechanical Engineering and Applied Mechanics, The University of Michigan, Ann Arbor, MI, 1995.

[29] Hajela, P., and Berke, L., "Neural Networks in Structural Analysis and Design: An Overview," *Computing Systems in Engineering*, Vol. 3, No. 1-4, pp. 525-538, 1992.

[30] Goel, S., and Hajela, P., "Identification of Couplings in Complex Engineering Design Problems Using Neural Networks," proceedings of the 5th AIAA/NASA/Air Force Symposium on Multidisciplinary Analysis and Optimization, Panama City, FL, September, 1994.

[31] Goldberg, D. E., Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley Publishing Co. Inc., Reading, Massachusetts, 1989.

[32] Hajela, P., and Lee, J., "Role of Emergent Computing Techniques in Multidisciplinary Rotor Blade Design," proceedings of the NATO Advanced Research Workshop on Emergent Computing Methods in Engineering Design, Grierson, D., and Hajela, P. (eds.), Nafplio, Greece, August 1994.

[33] Bauchau, O. A., and Kang, N. K., "A Multibody Formulation for Helicopter Structural Dynamic Analysis," *Journal of American Helicopter Society*, Vol. 38, No. 2, pp. 3-14, April 1993.